UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/996,088	11/28/2001	Peter C. Damron	004-7044	2526

22120 7590 10/18/2004

ZAGORIN O'BRIEN & GRAHAM, L.L.P.
7600B N. CAPITAL OF TEXAS HWY.
SUITE 350
AUSTIN, TX 78731

EXAMINER	
YIGDALL, MICHAEL J	

ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 10/18/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/996,088	DAMRON ET AL.	
	Examiner	Art Unit	
	Michael J. Yigdall	2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 November 2001*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-49* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-49* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *28 November 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 2/10/03, 4/11/03, 4/22/03
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-49 are pending and have been examined. The priority date considered for the application is November 28, 2001.

Specification

2. The abstract of the disclosure is objected to because the abstract must not exceed 150 words. Correction is required. See MPEP § 608.01(b).

3. Applicant is requested to update the status of the U.S. patent applications listed in the CROSS-REFERENCE TO RELATED APPLICATIONS section.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 1-32, 34, 35 and 37-49 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Pub. No. 2003/0074653 to Ju et al. (hereinafter "Ju").

With respect to claim 1, Ju discloses, in a scheduler for computer code wherein certain operations are likely to stall execution of the computer code and thereby provide latency for

completion of one or more pre-executable operations, a method of scheduling certain of the operations (see the abstract), the method comprising:

(a) for one or more sequences of operations that follow a speculation boundary and that define respective dependency chains, including pre-executable operations, which lead to likely stalls, representing speculative copies thereof as duplicate chains (see FIG. 1 and paragraph 0015, lines 1-16, which shows a sequence of operations following a slice boundary or speculation boundary that affect, i.e. define a dependency chains for, performance-degrading instructions that lead to likely stalls; see TABLE 1 and paragraph 0021, lines 1-19, which shows representing copies of the operations as backward slices for speculative execution; and see paragraph 0019, lines 1-4, which shows pre-executing the backward slice); and

(b) scheduling operations of the computer code, wherein the scheduling of operations from the duplicate chains is performed without regard to dependence of respective original operations on the speculation boundary, thereby scheduling certain of the operations above the speculation boundary into position preceding at least one of the operations likely to stall execution of the computer code (see paragraph 0030, lines 3-10, which shows issuing or scheduling operations from the speculative backward slices earlier than the performance-degrading instructions likely to stall, and paragraph 0024, lines 1-8, which shows scheduling without regard to the memory dependence).

With respect to claim 2, Ju further discloses the limitation wherein the likely stalls include likely cache misses (see paragraph 0016, lines 5-8, which shows that the performance-degrading likely stalls include likely cache misses).

With respect to claim 3, Ju further discloses the limitation wherein the dependency chains include address chains leading to memory access operations likely to miss in a cache (see paragraph 0018, lines 1-13, which shows backward slices defining dependency chains that include memory or address dependencies, i.e. address chains, leading to performance-degrading instructions, such as likely cache misses).

With respect to claim 4, Ju further discloses the limitation wherein the pre-executable operations include prefetch instructions (see paragraph 0022, lines 11-15, which shows executing the operations as prefetch instructions).

With respect to claim 5, Ju further discloses the limitation wherein the pre-executable operations include speculative operations (see paragraph 0023, lines 1-10, which shows executing the operations speculatively).

With respect to claim 6, Ju further discloses the limitation wherein the operations likely to stall execution include memory access instructions (see paragraph 0024, lines 1-8, which shows that the operations include memory access instructions).

With respect to claim 7, Ju further discloses the limitation wherein the operations likely to stall execution include operations selected from the set of:

(a) a load operation;

(b) first use of a load operation;

(c) a store operation;

(d) a branch operation;

(e) a multi-cycle computational operation;

(f) an iterative or recursive operation;

(g) a communications operation;

(h) an input/output (I/O) operation;

(i) a synchronization operation; and

(j) a co-processor operation

(see paragraph 0016, lines 5-8, which shows that the performance-degrading instructions likely to stall include branch operations).

With respect to claim 8, Ju further discloses the limitation wherein the speculation boundary is defined by one of:

(a) a store operation;

(b) a branch operation;

(c) a join operation;

(d) an iterative or recursive operation;

(e) a communications operation;

(f) an input/output (I/O) operation;

(g) a synchronization operation; and

(h) a co-processor operation

(see paragraph 0015, lines 9-13, which shows that the slice boundary or speculation boundary is defined by a synchronization operation).

With respect to claim 9, Ju further discloses inserting the pre-executable operations into the computer code (see paragraph 0030, lines 1-13, which shows inserting the pre-executable operations into the computer code).

With respect to claim 10, Ju further discloses profiling the computer code to identify the likely stalls (see paragraph 0016, lines 5-8, which shows identifying performance-degrading instructions likely to stall by profiling the computer code).

With respect to claim 11, Ju further discloses upon reaching the speculation boundary, deleting unscheduled operations of the duplicate chains and continuing to schedule respective original operations (see paragraph 0040, lines 9-13, which shows reaching the boundary of the speculative backward slice and terminating its execution, i.e. deleting its unscheduled operations and continuing to schedule the original operations).

With respect to claim 12, Ju further discloses deleting from the original operations, pre-executable operations for which a respective speculative copy is scheduled (see TABLE 3 and paragraph 0025, lines 1-8, which shows pre-executing a speculative operation instead of the original operation, i.e. deleting the pre-executable operation from the original operations).

With respect to claim 13, the limitations recited in the claims are analogous to the limitations of claim 1 (see Ju as applied to claim 1 above).

With respect to claim 14, the limitations recited in the claims are analogous to the limitations of claim 1 (see Ju as applied to claim 1 above).

With respect to claim 15, the limitations recited in the claim are analogous to the limitations of claim 3 (see Ju as applied to claim 3 above).

With respect to claim 16, the limitations recited in the claim are analogous to the limitations of claim 11 (see Ju as applied to claim 11 above).

With respect to claim 17, the limitations recited in the claim are analogous to the limitations of claim 7 (see Ju as applied to claim 7 above).

With respect to claim 18, the limitations recited in the claim are analogous to the limitations of claim 8 (see Ju as applied to claim 8 above).

With respect to claim 19, Ju further discloses the limitation wherein the speculation boundary is defined by an operation that has irreversible side-effects (see paragraph 0033, lines 1-2 and paragraph 0035, lines 1-11, which shows defining a slice boundary or speculation boundary at the point at which all the live-in variables are ready, i.e. at the operation at which irreversible side-effects are complete).

With respect to claim 20, the limitations recited in the claim are analogous to the limitations of claim 6 (see Ju as applied to claim 6 above).

With respect to claim 21, Ju further discloses, for at least load-type ones of the operations, inserting corresponding prefetch operations (see paragraph 0023, lines 1-10, which shows inserting advanced loads or prefetch operations for load operations).

With respect to claim 22, Ju further discloses converting load-type ones of the scheduled operations to speculative counterpart operations (see paragraph 0023, lines 1-10, which shows converting load operations to speculative load operations).

With respect to claim 23, Ju further discloses converting load-type ones of the scheduled operations to non-faulting loads (see paragraph 0023, lines 1-10, which shows converting load operations to advanced loads and performing a check rather than faulting).

With respect to claim 24, Ju further discloses, responsive to the scheduling of a prefetch operation from one of the duplicate dependency chains, disposing of a corresponding prefetch operation from a corresponding one of the original dependency chains (see paragraph 0022, lines 11-15, which shows scheduling a speculative prefetch operation from the backward slice instead of from the original operations, i.e. disposing of the original operation).

With respect to claim 25, Ju further discloses selecting for the scheduling, particular ones of the operations from the duplicate dependency chains based at least in part on chain length (see paragraph 0019, lines 1-9, which shows selecting operations for the backward slice according to the size of the slice, i.e. the chain length).

With respect to claim 26, the limitations recited in the claim are analogous to the limitations of claim 2 (see Ju as applied to claim 2 above).

With respect to claim 27, the limitations recited in the claim are analogous to the limitations of claim 2 (see Ju as applied to claim 2 above).

With respect to claim 28, Ju further discloses the limitation wherein the likely to stall operations include operations that stall an execution pipeline (see paragraph 0039, lines 1-9, which shows operations that stall an execution pipeline).

With respect to claim 29, Ju further discloses the limitation wherein the dependency chains include load-type and prefetch operations (see paragraph 0024, lines 1-8, which shows that the dependency chains include loads and advanced load or prefetch operations).

With respect to claim 30, Ju further discloses the limitation wherein the dependency chains include operations other than load-type and prefetch operations (see TABLE 2 and paragraph 0024, lines 1-8, which shows that the dependency chains include other operations).

With respect to claim 31, Ju further discloses the limitation wherein the dependency chains include operations involved in address calculations (see TABLE 2 and paragraph 0024, lines 1-8, which shows that the dependency chains include operations for calculating addresses of memory locations).

With respect to claim 32, Ju further discloses the limitation wherein the duplicate dependency chains are represented as copies of the respective original dependency chains with speculation boundary dependencies removed or ignored (see paragraph 0023, lines 1-10, which shows that the backward slice includes speculative copies of the original operations, and paragraph 0024, lines 1-8, which shows that the dependencies are ignored).

With respect to claim 34, Ju further discloses the limitation wherein the dependencies include one or more of:

(a) register dependencies;

(b) branch dependencies; and

(c) memory dependencies

(see paragraph 0018, lines 1-13, which shows that the dependencies include memory dependencies).

With respect to claim 35, Ju further discloses the limitation wherein the method is realized in an optimizing compiler (see paragraph 0014, lines 1-4, which shows a compiler for resolving long latency events, i.e. an optimizing compiler).

With respect to claim 37, the limitations recited in the claim are analogous to the limitations of claims 1 and 3 (see Ju as applied to claims 1 and 3 above).

With respect to claim 38, Ju further discloses encoding the scheduled operations as part of the program code (see paragraph 0014, lines 4-10, which shows encoding the scheduled instructions as part of the program code).

With respect to claim 39, the limitations recited in the claim are analogous to the limitations of claim 2 (see Ju as applied to claim 2 above).

With respect to claim 40, the limitations recited in the claim are analogous to the limitations of claims 1 and 4 (see Ju as applied to claims 1 and 4 above).

With respect to claim 41, the limitations recited in the claim are analogous to the limitations of claim 2 (see Ju as applied to claim 2 above).

With respect to claim 42, Ju discloses a computer program product encoded in one or more computer readable media (see paragraph 0038, lines 4-6), the computer program product comprising an execution sequence of instructions, the execution sequence including subsequence that includes a speculative load instruction that feeds a subsequent prefetch instruction (see FIG. 1 and paragraph 0015, lines 1-16, which shows an execution sequence of instructions and a speculative backward slice or sub-sequence; see paragraph 0023, lines 1-10, which shows advanced or speculative loads; and see paragraph 0022, lines 11-15, which shows prefetching data for a performance-degrading instruction).

With respect to claim 43, Ju further discloses one or more instructions disposed between the speculative load instruction and the subsequent prefetch instruction in the execution sequence (see TABLE 1 and paragraph 0021, lines 1-19, which shows an instruction between the speculative load instruction I5 and the subsequent prefetch instruction I7).

With respect to claim 44, Ju further discloses a martyr instruction that follows the speculative load instruction and the prefetch instruction which, upon execution, provides at least a portion of a latency therefor (see TABLE 1 and paragraph 0021, lines 1-19, which shows the performance-degrading instruction I7 following the speculative load instruction I5 and the prefetch execution of I7, and paragraph 0014, lines 1-4, which shows that the performance-degrading instruction provides latency).

With respect to claim 45, Ju further discloses the limitation wherein the computer program product is prepared by a program scheduler that inserts prefetch instructions into the

execution sequence and schedules speculative duplicates of at least some load instructions together with corresponding prefetch instructions above speculative boundaries therein (see paragraph 0030, lines 1-13, which shows inserting prefetch instructions into the execution sequence and scheduling a speculative backward slice above the termination boundary or speculative boundary).

With respect to claim 46, Ju further discloses the limitation wherein the one or more computer readable media are selected from the set of a disk, tape or other magnetic, optical, semiconductor or electronic storage medium and a network, wire line, wireless or other communications medium (see paragraph 0013, lines 1-19).

With respect to claim 47, Ju discloses an apparatus comprising:

(a) a code preparation facility for transforming schedulable code into scheduled code (see paragraph 0030, lines 1-13, which shows a code preparation facility for scheduling code); and

(b) means for scheduling speculative copies of operations that form dependency chains that lead to a likely stall, the scheduling placing the speculative operations above a preceding at least one other operation that is itself likely to stall, thereby hiding in the scheduled code latency of the speculative operations (see FIG. 1 and paragraph 0015, lines 1-16, which shows a backward slice, i.e. operations that form dependency chains, leading to a performance-degrading instruction or likely stall; see paragraph 0030, lines 3-10, which shows scheduling the speculative backward slice above the performance-degrading instruction; and see paragraph 0035, lines 14-17, which shows hiding the latency of the performance-degrading instruction).

With respect to claim 48, Ju further discloses means for inserting pre-executable operations into the schedulable code, wherein at least some of the pre-executable operations are scheduled by the scheduling means as the speculative operations for which latency is hidden (see paragraph 0019, lines 1-4, which shows pre-executable operations, and paragraph 0023, lines 1-10, which shows inserting pre-executable operations for speculative execution).

With respect to claim 49, Ju further discloses means for identifying likely-to-stall operations of schedulable code (see paragraph 0016, lines 5-8, which shows identifying performance-degrading instructions likely to stall).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 33 and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ju, as applied to claim 13 above, in view of U.S. Pub. No. 2002/0056078 to Inagaki et al. (hereinafter "Inagaki").

With respect to claim 33, although Ju discloses using known program analysis techniques (see paragraph 0030, lines 10-13), Ju does not expressly discloses the limitation wherein the dependency chains are represented in a directed acyclic graph of dependencies amongst the corresponding operations.

However, Inagaki discloses a program optimization method (see the abstract) wherein the dependencies are represented in a directed acyclic graph (DAG) (see paragraph 0088, lines 1-6). The DAG represents data dependencies, order restrictions and control dependencies among the operations (see paragraphs 0089-0091).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to represent the dependency chains of Ju in a DAG, as taught by Inagaki, for the purpose of identifying data dependencies, order restrictions and control dependencies when resolving performance-degrading events.

With respect to claim 36, although Ju discloses an optimizing compiler (see paragraph 0014, lines 1-4), Ju does not expressly disclose the limitation wherein the method is realized in a just-in-time (JIT) compiler.

However, Inagaki discloses a program optimization method (see the abstract) realized in a Java just-in-time compiler (see paragraph 0075, lines 1-12).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to realize the method of Ju in a just-in-time compiler, as taught by Inagaki, for the purpose of resolving performance-degrading events in programs written in the Java language.

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to Applicant's disclosure. U.S. Pat. No. 6,697,932 to Yoaz et al. discloses a system and method for early resolution of low confidence branches and safe data cache accesses. U.S. Pat. No. 6,675,374 to Pieper et al. discloses the insertion of prefetch instructions into computer program code. U.S.

Pat. No. 6,748,589 to Johnson et al. discloses a method for increasing the speed of speculative execution. U.S. Pat. No. 6,098,166 to Liebholz et al. discloses the speculative issue of instructions under a load miss shadow. U.S. Pat. No. 5,948,095 to Arora et al. discloses a method and apparatus for prefetching data in a computer system.

9. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael J. Yigdall whose telephone number is (703) 305-0352. The examiner can normally be reached on Monday through Friday from 7:30am to 4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (703) 305-4552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

After October 25, 2004, the examiner can be reached at (571) 272-3707, and the examiner's supervisor, Tuan Q. Dam can be reached at (571) 272-3695.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MY

Michael J. Yigdall
Examiner
Art Unit 2122

mjy

TUAN DAM
SUPERVISORY PATENT EXAMINER